

# UnicodeToBytes

Poses a risk of buffer overflow if the return buffer is not appropriately sized.

Sean Barnum, Cigital, Inc. [vita<sup>1</sup>]

Copyright © 2007 Cigital, Inc.

2007-04-23

## Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 6228 bytes

<b>Attack Category</b>	<ul style="list-style-type: none"><li>• Malicious Input</li></ul>						
<b>Vulnerability Category</b>	<ul style="list-style-type: none"><li>• Multibyte Character</li><li>• Buffer Overflow</li><li>• Unconditional</li></ul>						
<b>Software Context</b>	<ul style="list-style-type: none"><li>• String Conversion MACROS</li><li>• National Language Support</li></ul>						
<b>Location</b>	<ul style="list-style-type: none"><li>• GB18030.h</li></ul>						
<b>Description</b>	<p>UnicodeToBytes() poses a risk of buffer overflow if the return buffer is not appropriately sized.</p> <p>UnicodeToBytes() converts Unicode characters to GB18030 bytes (where GB18030 is the standard multibyte character set for the Chinese language). Using the UnicodeToBytes function incorrectly can compromise the security of your application. To avoid a buffer overrun, be sure to specify a buffer size appropriate for the data type the buffer receives. It is easy to accidentally use the wrong units to specify buffer size, since Unicode size is specified as a number of characters and multibyte size is specified as a number of bytes.</p> <p>Use of WideCharToMultiByte is preferred over UnicodeToBytes.</p>						
<b>APIs</b>	<table border="1"><thead><tr><th>Function Name</th><th>Comments</th></tr></thead><tbody><tr><td>UnicodeToBytes</td><td></td></tr></tbody></table>	Function Name	Comments	UnicodeToBytes			
Function Name	Comments						
UnicodeToBytes							
<b>Method of Attack</b>	BufferOverflow						
<b>Exception Criteria</b>							
<b>Solutions</b>	<table border="1"><thead><tr><th>Solution Applicability</th><th>Solution Description</th><th>Solution Efficacy</th></tr></thead><tbody><tr><td>Whenever converting Unicode to GB18030.</td><td>Prefer use of WideCharToMultiByte() over use ofUnicodeToBytes().</td><td>Effective.</td></tr></tbody></table>	Solution Applicability	Solution Description	Solution Efficacy	Whenever converting Unicode to GB18030.	Prefer use of WideCharToMultiByte() over use ofUnicodeToBytes().	Effective.
Solution Applicability	Solution Description	Solution Efficacy					
Whenever converting Unicode to GB18030.	Prefer use of WideCharToMultiByte() over use ofUnicodeToBytes().	Effective.					

1. [http://buildsecurityin.us-cert.gov/bsi/about\\_us/authors/35-BSI.html](http://buildsecurityin.us-cert.gov/bsi/about_us/authors/35-BSI.html) (Barnum, Sean)

	<p>If UnicodeToBytes() must be used</p>	<p>To ensure that the size of the lpMultiByteStr (Destination) buffer is large enough contain the converted string, you should call UnicodeToBytes twice -- once to determine the size of the translated string and again to translate the string.</p>	<p>Effective.</p>
	<p>If UnicodeToBytes() must be used</p>	<p>Make sure that size of Unicode is specified in characters while size of multibyte text is specified in bytes.</p>	<p>Effective.</p>
<p><b>Signature Details</b></p>	<p>DWORD UnicodeToBytes( LPWSTR lpWideCharStr, UINT cchWideChar, LPSTR lpMultiByteStr, UINT cchMultiByte );</p>		
<p><b>Examples of Incorrect Code</b></p>	<pre> LPWSTR szWideCharStr = L"Some Unicode text to be converted. Pretend exotic characters are included."; DWORD cchMultiByte = sizeof(szWideCharStr); // Wrong - might not be large enough LPSTR lpMultiByteStr = new CHAR[ cchMultiByte + 1 ]; // Allocate block for return UnicodeToBytes( szWideCharStr, // Unicode string to be converted [in] sizeof(szWideCharStr), // Wrong! give number of bytes when number WCHAR is needed lpMultiByteStr, cchMultiByte ); // </pre>		

<b>Examples of Corrected Code</b>	<pre> /* If one must use UnicodeToBytes(), do the following. But note that use of WideCharToMultiByte() is preferred. */  UINT cchWideCharConvertCount = lstrlenW(szWideCharStr); // Convert all characters in szWideCharStr UINT cchMultiByte = UnicodeToBytes( szWideCharStr, // Unicode string to be converted [in] cchWideCharConvertCount, NULL, 0 ); // Return byte count of conversion LPSTR lpMultiByteStr = new CHAR[ cchMultiByte + 1 ]; // Allocate block for return UnicodeToBytes( szWideCharStr, // Unicode string to be converted [in] cchWideCharConvertCount, lpMultiByteStr, cchMultiByte );  /* Note: error checking has been omitted for brevity */ </pre>				
<b>Source Reference</b>	<ul style="list-style-type: none"> <li>• <a href="http://msdn.microsoft.com/library/default.asp?url=/library/en-us/intl/unicode_UnicodeToBytes.asp">http://msdn.microsoft.com/library/default.asp?url=/library/en-us/intl/unicode_UnicodeToBytes.asp</a><sup>2</sup></li> </ul>				
<b>Recommended Resources</b>	<ul style="list-style-type: none"> <li>• <a href="#">MSDN reference for UnicodeToBytes</a><sup>3</sup></li> <li>• <a href="#">MSDN Reference for WideCharToMultiByte</a><sup>4</sup></li> </ul>				
<b>Discriminant Set</b>	<table border="1"> <tr> <td data-bbox="807 1370 1114 1415"><b>Operating System</b></td> <td data-bbox="1120 1370 1422 1415"> <ul style="list-style-type: none"> <li>• Windows</li> </ul> </td> </tr> <tr> <td data-bbox="807 1424 1114 1507"><b>Languages</b></td> <td data-bbox="1120 1424 1422 1507"> <ul style="list-style-type: none"> <li>• C</li> <li>• C++</li> </ul> </td> </tr> </table>	<b>Operating System</b>	<ul style="list-style-type: none"> <li>• Windows</li> </ul>	<b>Languages</b>	<ul style="list-style-type: none"> <li>• C</li> <li>• C++</li> </ul>
<b>Operating System</b>	<ul style="list-style-type: none"> <li>• Windows</li> </ul>				
<b>Languages</b>	<ul style="list-style-type: none"> <li>• C</li> <li>• C++</li> </ul>				

## Cigital, Inc. Copyright

Copyright © Cigital, Inc. 2005-2007. Cigital retains copyrights to this material.

Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and “No Warranty” statements are included with all reproductions and derivative works.

For information regarding external or commercial use of copyrighted materials owned by Cigital, including information about “Fair Use,” contact Cigital at [copyright@cigital.com](mailto:copyright@cigital.com)<sup>1</sup>.

1. <mailto:copyright@cigital.com>

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.